Emergent Properties from Feature Co-occurrence in Image Collections

Umair Mateen Khan, Steven Mills, Brendan McCane, and Andrew Trotman
Department of Computer Science, University of Otago, Dunedin, New Zealand
{umairkhan, steven, mccane, andrew}@cs.otago.ac.nz

***Abstract*—This paper proposes a novel approach to explore emergent patterns in images in an unsupervised setting. We consider emergent patterns to be sets of co-occurring visual words that appear together more often than chance would indicate. Rather than focusing on finding ways to learn a large number of objects or their categories we focus on analyzing behavior associated with emergent patterns. We show that these patterns emerge from the data and in some cases relate to object identifiers. We extract SIFT descriptors [1] and then cluster them to represent each image as a bag-of-words. To encode co-occurrences between visual words we represent them as edges of a graph which are weighted by the number of images containing a particular co-occurrence. Performing a statistical analysis on weights of the edges identifies words which co-occur significantly more often than expected. These highly co-occurring nodes produce clusters in the graph which can be separated using normalized cuts. Applying normalized cuts reveals that in simple images datasets these emergent clusters can identify object classes. Results on more complex datasets like Caltech101 [2] show that interesting patterns other than object classes can also emerge from the data.**

I. Introduction

Learning useful information from data in an unsupervised way is a challenging task but is becoming increasingly important given the need to learn from extremely large datasets [3], [4], [5], [6]. The main motivation for such approaches over supervised ones is scalability, which is a prerequisite when working with large image datasets. We hypothesize that emergent properties can serve as a basis for machine vision systems even though such properties may be different to those humans use.

In this paper we propose an approach for exploring emergent patterns in a set of images in an unsupervised manner. Our approach is inspired by graph theoretic approaches and represents co-occurrence information among words in the form of a large graph. Partitioning this graph results in clusters of co-occurring words which emerge from the dataset. Evaluating emergent information is a challenging task, so to see whether our proposed approach is finding useful emergent information, a simpler dataset is initially used. Later on experiments on a more complex dataset are conducted.

The main contribution of this paper is to suggest a novel criterion for choosing significant edges of the graph which are used in the graph partitioning process. This allows us to address the question — what do emerging patterns look like in databases of different levels of complexity? While these patterns may correspond to our *a priori* expectations of object categories or classes, this need not be the case.

Related Work

Prior literature on unsupervised learning from images can be categorized into 3 broad domains: itemset-mining, statistical and graph-based.

In [7] knowledge discovery in databases (KDD) was introduced and later several algorithms such as Apriori [8], Eclat [9], Fp-Growth [10] and Apriori Inverse [11] were developed to mine information from market basket datasets generated from list of items purchased by customers in their groceries. The grocery items in a basket or a transaction are called itemset. These algorithms focused on finding frequent and rare itemset from the entire datasets.

Quack et al. [12] use itemset mining techniques, such as Apriori [8], for solving computer vision problems. Each local neighborhood of image features is considered as a transaction in a market-basket metaphor. Frequent itemset mining techniques are applied and result in frequently co-occurring features pertaining to particular objects or movie actors. Similarly Malik [13] uses association rule mining for clustering web images. Association rules are generated for both visual and text features from web pages and then represented this as hypergraphs, which are clustered to find similar images. Several other papers have also used similar data mining approaches to find interesting patterns in images [14], [15], [16], [17].

Sivic et al. [3] use statistical modeling to discover patterns in images. They use topic modeling via probabilistic Latent Semantic Analysis (pLSA). Topic modeling is a kind of generative model initially used for identifying topics in text corpora and assumes each document has a mixture of topics, and each topic has a distribution of words. They use this approach for topic discovery from unlabelled images; classification of unseen images; and object detection. Performance was evaluated on four categories (faces, motorbikes, airplanes, cars) from the Caltech101 dataset, giving an accuracy of more than 90%. They also introduce the notion of doublets which store information about spatially co-occurring regions. In another work Sivic et al. [18] introduce the hierarchical Latent Dirichlet Allocation (hLDA) model for automatically discovering object hierarchies from unlabeled data. Other notable work that uses statistical modeling based approaches includes [5], [19].

Another broad area uses graph theoretic approaches to mine information. These techniques build graphs from visual words or images by capturing their correspondences. Graph clustering or edge cutting techniques are then applied to obtain partitions that represent objects. Grauman and Darrell [20] present an approach for categorizing unlabelled images. Their approach starts by finding feature correspondences and based on these

correspondences they calculate affinities among images. A spectral clustering technique is then applied to partition data. The partitioned data can be used to train classifiers for different object categories. Evaluation on the Caltech-4 dataset gives 94% accuracy when grouping images according to their categories. In another approach Kim et al [21] created a large graph directly from image features and then captured object category information using analysis techniques widely used in web and social network data. For the 4 objects from the Caltech101 dataset they achieved a classification accuracy of 95.42% and for 3 objects from TUD/ETHZ dataset a classification accuracy of 95.47% was achieved.

Most previous work described focuses on unsupervised discovery of object categories which is different from what we aim for in our approach. We focus on finding emergent patterns using co-occurrence of features that might or might not relate to a single object.

II. Proposed Approach

We consider the problem of finding emergent patterns as a graph clustering problem where clusters in a graph represent some emergent behavior. The graph contains co-occurrence information among different visual words. An edge weight represents how many images contain a pair of words that co-occur. The goal is to partition the graph into parts that represent emergent patterns.

In this paper we suggest an approach to select significant edges (co-occurrences) from a large numbers of total edges. Keeping only significant edges not only reduces the complexity of the graph partitioning process but also facilitates emergent patterns by only keeping the most important information.

A. Co-occurrence Graph

We use SIFT [1] descriptors to represent image features. These descriptors are then quantized by using K-Means to generate a Bag of Words representation for each image. For our approach co-occurrence is the basis for emergence. Co-occurrence is encoded in a graph $G(V, E)$ where a node, $v_i \in V$, represents a single visual word, and an edge, $e_i \equiv (v_i, v_j, f_{ij}) \in E$ encodes the frequency, f_{ij}, of co-occurrence of two visual words, v_i and v_j, in the image set (ignoring multiple co-occurrences in a single image). The resulting graph is an undirected weighted graph.

B. Statistical Analysis and Graph Visualization

Significant edges are chosen based on the binomial test. The null hypothesis is that visual words appear independently in each image. Those words that co-occur much more frequently than expected are dependent on each other and are result of some activity in images and hence they are chosen as significant co-occurrences. This is measured by calculating a z-score for each co-occurrence or an edge in our graph. Details of this process is given in Algorithm 1. This z-score is also assigned as edge weight by replacing previous co-occurrence count based weight. The edges are then thresholded based on this value, in order to keep only the most significant co-occurrences. We experimented with multiple z-score values as thresholds discussed in Sub-section III-A.

Algorithm 1: Finding significant edges

Data: A graph $G = (V, E)$ where an edge is a co-occurrence of two words and edge weight encodes the frequency of this co-occurrence
Result: Edges score according to their statistical significance

for *each* $v_i \in V$ **do**
  Find probability of each vertex: $P(v_i) = \frac{\text{\# of document containing this word}}{\text{total documents (n)}}$
for *each edge* $e_k \in E$ *in the graph* **do**
  Compute joint probability of vertices v_i and v_j co-occurring together $P(v_i v_j) = P(v_i)P(v_j)$
  Compute the probability of failure $Q(v_i v_j) = 1 - P(v_i v_j)$
  Compute the mean of the binomial distribution $\mu = nP(v_i v_j)$
  Compute the standard deviation: $\sigma = \sqrt{nP(v_i v_j)Q(v_i v_j)}$
  z-score $= \frac{x - \mu}{\sigma}$
  weight of an edge$(w_{ij}) = z$-score

The graph obtained after applying a threshold value is visualized using the graph viewing package Gephi [22]. Visualizing the graph allows us to see emergent clusters before analyzing them using a graph partitioning or clustering method. The edge weight distribution can also give important insight to the graph structure and could also be used for deciding the edge threshold value. As described earlier, emergent patterns appear as dense clusters of words in the graph, ideally disconnected from each other (although this is not the typical case).

C. Graph Clustering

Graphs obtained after only keeping the most significant edges still have many connections among emergent clusters. In order to obtain vertices that are related to each cluster we use spectral clustering via normalized cuts [23]. Unlike other graph partitioning techniques, spectral clustering uses the spectrum (eigenvalues) of an affinity matrix of data points to obtain a dimensionality reduced version of the data. The affinity matrix W is defined as:

$$W_{ij} = \begin{cases} w_{ij}, \ if \ (v_i, v_j) \in E \\ 1, \ if \ i = j \\ 0, \ otherwise. \end{cases} \quad (1)$$

Here w_{ij} is the weight (z-score) of the edge calculated during statistical analysis.

Experiments were performed with two partitioning algorithms: recursive two-way Ncut and simultaneous K-way cut with multiple eigenvectors as described in [23]. The recursive two-way Ncut approach is a hierarchical partitioning technique, creating a tree of partitions. The root of this tree contains all nodes of the graph. At each level a partitioning problem $(D - W)x = \lambda Dx$ is solved for the eigenvectors with the K smallest eigenvalues. Here W is the affinity matrix of the graph and D is a diagonal matrix containing the sum

of the weights incident on each vertex. For this approach, only the second smallest eigenvalue is used to bipartition the graph, although the next few eigenvectors also contain useful partitioning information. For each subgraph we again solve the partitioning problem and similarly use the second smallest eigenvalue to recursively partition it down to a fixed level. This approach can be computationally wasteful, since it only uses the second smallest eigenvector and ignores all others. For the Simultaneous K-way cut approach, instead of using the second smallest eigenvalues, the top K eigenvectors are used to get a K-way partition [23].

Purity Measure: Partitions obtained after normalized cuts split words into different emergent clusters. In a scenario where a dataset contains images related to some categories of objects, it is possible that an emergent cluster may represent one of these objects categories. To compute how much, or to what extent words from an emergent cluster are related to a particular object category, a metric called $purity$ is used.

The $purity$ of a cluster can be defined as:

$$purity(i) = max_j(P_i(c_j)), \text{ and } \; i = 1 \text{ to } K \qquad (2)$$

Here, K is the total number of clusters obtained after normalized cuts and all the words in these clusters are from C categories of objects. Also, $P_i(c_j)$ is the proportion of the words from the j_{th} category to the total number of the words in all other categories, found in the i_{th} cluster. The $purity$ of a cluster can only help us to evaluate whether a particular cluster belongs to any object category or not and does not help to find emergent clusters. There could be a perfectly valid emergent cluster that does not belong to a single object category.

D. Image Ranking

Once we have different partitions or emergent clusters containing sets of words that co-occurred significantly more than their expectation, it is desirable to visualize these clusters. All images containing these clusters are identified and locations of words in them are marked. A ranking mechanism is also presented to rank images according to the number of significant edges they contain. If an image contains a large number of significant edges from a cluster then it covers a majority of the co-occurrences from this cluster and hence, is a better representation of this cluster. The total number of significant co-occurrences present in an image gives the rank of an image.

III. Experimental Setup and Results

We have experimented with two different image datasets. In the first experiment we investigate emergence in a very simple set of images. We created a dataset of 6000 images named Toymix as shown in Figure 6(a). It contains 6 objects and for each object there are exactly 1000 images. This dataset has low complexity as there are only small transformations (rotations, translations and scale changes) in it and all images have a plain background. To see emergence in a more complex scenario we chose the Caltech101 [6] dataset and used 20 categories from it, chosen based on a criteria of having at-least 80 images, in total 1600 images from 20 categories. We used, the first 80 images per object to give equal weighting to all categories even when more images were available. Figure 6(b) shows single images from each of the chosen categories. After extracting SIFT [1] descriptors from images we quantized them into 10,000 visual words in our experiments.

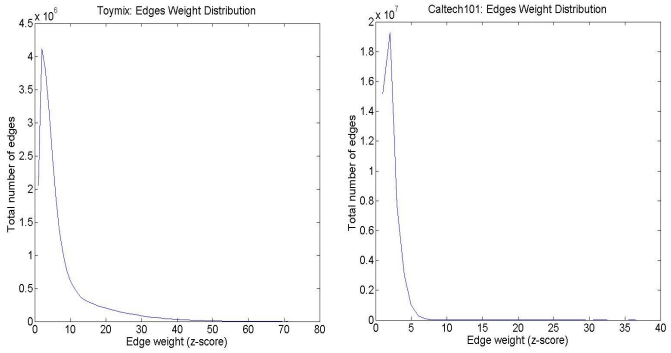

Figure 1. Edge weights distributions for both dataset for 10,000 visual words.

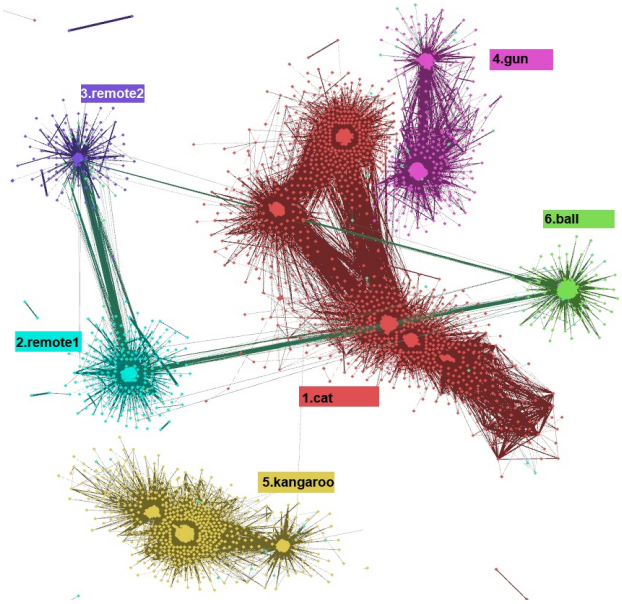

Figure 2. Graph structure obtained for 6 objects categories from the Toymix dataset. Here we keep significant edges by choosing a minimum z-score of 30 as a threshold of significance (These figures are best viewed in color)

A. Graph building and Significant Co-occurrences Selection

For the Toymix dataset there are approximately 26 million edges and for the Caltech101 dataset there are 46 million edges. Figure 1 shows the edge weight distribution for the graphs containing 10,000 vertices (visual word) generated for each dataset. The maximum edge weights are z-scores of 68 and 35 respectively. There are also significant differences in the shape of the distributions. In the Toymix case the z-scores are more spread out than in the Caltech101 case. As in the Toymix, a large number of non-unique co-occurrences resulted in having a large number of bins in the edge weight histogram. On the other hand, in Caltech101 there are fewer non-unique co-occurrences which caused fewer bins.

Edges that have high z-scores are more significant because such edges appeared more than expected and so should contain more important information. Up until this stage, no graph cutting or partitioning technique has been applied. The Toymix

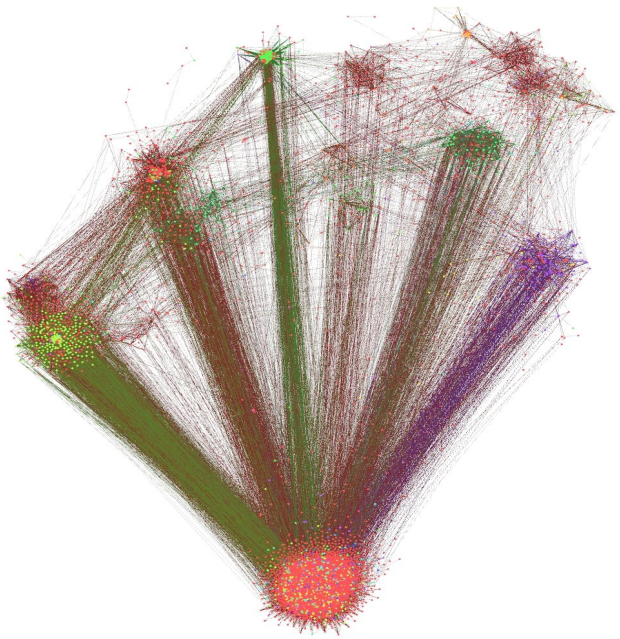

Figure 3. Graph structure obtained using 20 objects categories from Caltech101 dataset. Here we keep significant edges by choosing a minimum z-score of 4 as a threshold of significance (These figures are best viewed in color)

dataset graphs were visualized by setting threshold values of 20, 30 and 40 and for the Caltech101 dataset thresholds of 3, 4 and 5 are examined. For these threshold values, we have found similar graph structures. The resulting graphs are visualized in Gephi [22] using OpenOrd [24] layout. Figure 2 shows a graph generated from the Toymix dataset, and all edges having a z-score of more than 30 are displayed. Figure 3 depicts a graph for the Caltech101 dataset and contains all edges having a z-score of more than 4.

Vertices of these graphs are visual words and are color coded by assigning a label to each word according to their dominating object category. For each word its occurrence count in each object category $c_i \in C$ is computed. A word is assigned to the category it appears in most often. This is done by computing a ratio of the highest category count, to the second highest category count, $r = \frac{\text{Second highest category count}}{\text{Highest category count}}$. For a word if this ratio, $r < 0.6667$, then this word is assigned the category in which it mostly appeared. A default category $c_{def} \notin C$ is assigned in cases where this criterion is not meet.

The first graph, shown in Figure 2, is for the Toymix dataset and contains 6 object categories. When viewed using a graph drawing layout algorithm, six natural categories emerge, without applying any graph partitioning or edge cutting technique. Each emergent cluster is dominated by words from a single category as depicted by different colors for each cluster. These colors are the same as those assigned to the object categories in Figure 6(a). Most of the clusters are well separated, while the two remotes (which are visually similar) have many overlapping edges. This intuitively make senses because words in these clusters are very similar to each other.

The second graph, depicted in Figure 3, is for 20 categories of Caltech101 dataset. The graph has a complex structure but still there are some clusters, clearly having vertices dominated by a single category of object. These clusters are a lot more dense, but do not show a clear separation of different object categories because many linking edges remain. It is very interesting to see these clusters emerging before applying any graph clustering approach. As these clusters are not separated, assigning any category to them is not possible until these linking edges are removed. This is done by using a normalized cut algorithm.

B. Graph Clustering and Displaying Ranked Results

The clusters shown in Figures 2 and 3 still have edges between them. Spectral graph clustering is applied to these graphs to completely separate the clusters using methods described in Section II-C. For the simultaneous K-way cut approach K is selected as K = 6 and K = 20 for Toymix and Caltech101 dataset respectively. The reason for choosing these values is having the same number of object categories in each dataset even though emergent clusters may not correspond exactly to object categories. The recursive two-way approach is a hierarchical splitting and partitioning method where vertices at each level are split into two parts until a desired level L of this tree is reached. For both datasets, partitioning is applied recursively until a depth of $L = 16$ is attained, and creates a total of $2^{15} = 32,768$ partitions in total.

For the simultaneous K-way approach on the Toymix dataset, $K = 6$ is used as shown in Figure 4. The results show that, for each cluster, the top ranked images were from one of the object categories. For the Caltech101 dataset, setting K = 20 identifies six clusters, five of which relate to object categories as shown in Figure 5. We did not see any other semantic category emerging from the data apart from these 6. The reason for not seeing any other object category could be due to more variation among images from those categories. By looking at the last emergent category, the features in images are related to either backgrounds or heavily textured object for example sand or grass. It is interesting to see that this category represents an emergent property which is not directly related to any of the human-labeled object categories.

For the recursive two-way approach, images found in each cluster are stored only if they have at least ten visual words in them. Any cluster having fewer visual words are ignored. The top 200 images are saved for each cluster in separate directories. Emergent patterns are found related to object categories as we go down the tree. Some of the categories from the Toymix dataset: ball, remote1 and remote2 emerge in the upper and middle levels of the tree with a maximum $purity$ of 1. This means that the top 200 images in this cluster are related to a single category. Other categories emerge at deeper levels: Kangaroo, cat and gun categories all emerge at level 14 with the maximum $purity$ of 1. For the Caltech101 dataset a $purity$ of 0.45 was achieved when sunflower category emerged at level 14 of the tree. No other emergent clusters relating to object categories are observed using this approach.

IV. Conclusion

In this paper an approach to explore emergent patterns in image datasets is presented. The approach represents the co-occurrence of words from images in an undirected weighted

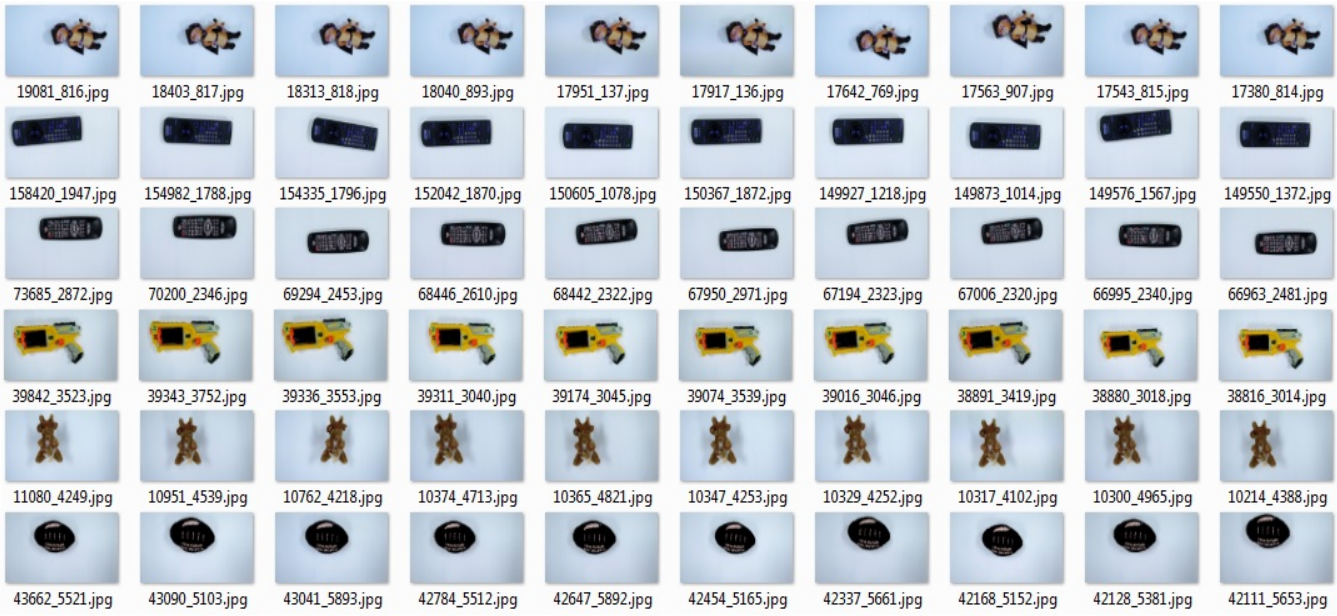

Figure 4. Six categories from Toymix dataset that emerged from data.Top 10 ranked images from each category are displayed.

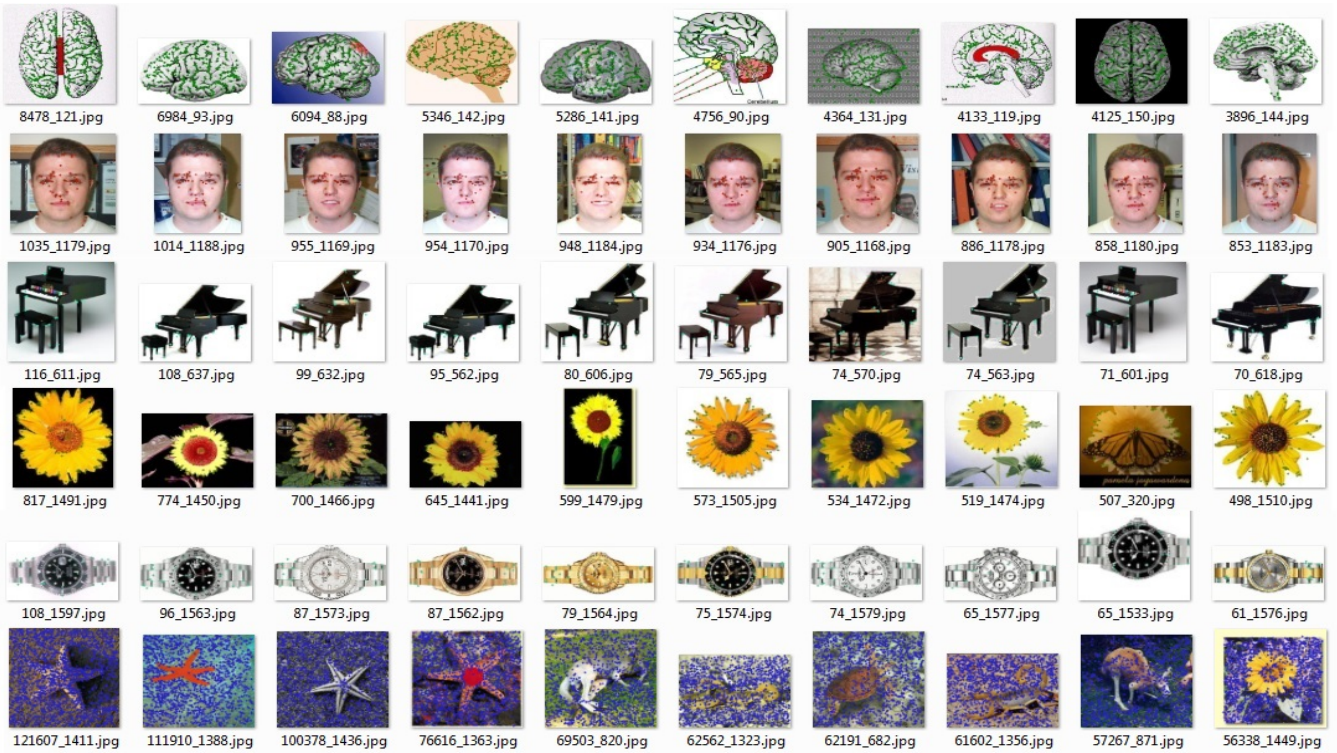

Figure 5. Five categories {Brain, Faces_easy, Piano, Sunflower, Watch and Background} from Caltech101 dataset that emerged from data. The top 10 ranked images from each category are displayed.

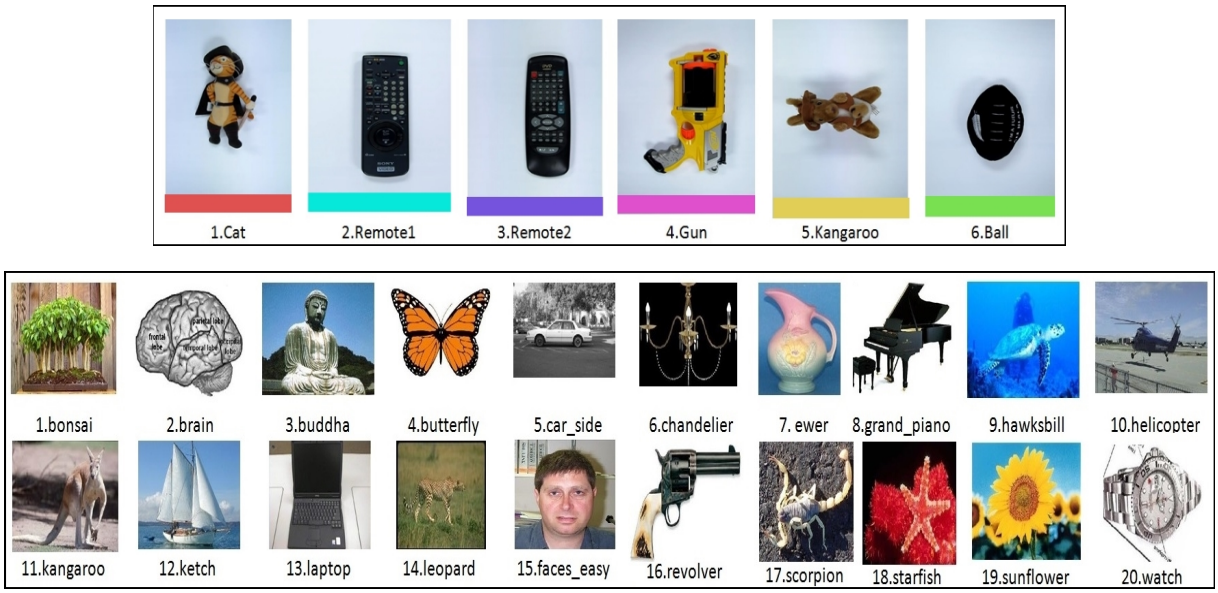

Figure 6. Datasets used in our experiments. The top row is the Toymix dataset and the color bars at the bottom of each image links each object to one of the clusters in the graph shown in Figure 2. The bottom image shows single images from 20 different categories from Caltech101 dataset.

graph. In this graph emergent patterns lead to dense clusters of nodes having high edge density. Identifying these clusters is still a challenging task. Applying our statistical criterion to these co-occurrences identifies emergent patterns from this data. We show that image datasets (e.g. 6 categories from Toymix) with low complexity can even result in class identifiers being found using our significance criteria. We also show that in a more challenging dataset like Caltech101 emergence can result in various interesting patterns including, but not limited to, some object categories.

ACKNOWLEDGMENT

The authors would like to thank Timothee Cour, Stella Yu and Jianbo Shi for their MATLAB Normalized Cuts Segmentation Code.

REFERENCES

[1] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," 2004.

[2] R. P. L. Fei-Fei; Fergus, "One-shot learning of object categories," *IEEE Transactions on Pattern Analysis Machine Intelligence*, 2006.

[3] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman, "Discovering objects and their location in images," 2005.

[4] A. Coates, A. Karpathy, and A. Y. Ng, "Emergence of object-selective features in unsupervised feature learning." in *NIPS*, 2012.

[5] D. Liu and T. Chen, "Unsupervised image categorization and object localization using topic models and correspondences between images," *ICCV*, 2007.

[6] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories." in *Workshop on Generative-Model Based Vision*, 2004.

[7] U. Fayyad, G. Piatetsky-shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," 1996.

[8] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules in large databases," in *VLDB'94*. Morgan Kaufmann, 1994.

[9] M. J. Zaki, S. Parthasarathy, M. Ogihara, and W. Li, "New algorithms for fast discovery of association rules," in *3rd International Conference on Knowledge Discovery and Data Mining (KDD)*, 1997.

[10] J. Han, J. Pei, Y. Yin, and R. Mao, "Mining frequent patterns without candidate generation: A frequent-pattern tree approach," 2004.

[11] Y. S. Koh and N. Rountree, "Finding sporadic rules using apriori-inverse," in *Proceedings of the 9th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*. Springer-Verlag, 2005.

[12] T. Quack, V. Ferrari, and L. V. Gool, "Video mining with frequent itemset configurations," in *CIVR*. Springer, 2006.

[13] H. H. Malik, "Clustering web images using association rules, interestingness measures, and hypergraph partitions," in *Proceedings of the 6th international conference on Web engineering*. ACM Press, 2006.

[14] T. Quack, V. Ferrari, B. Leibe, and L. J. V. Gool, "Efficient mining of frequent and distinctive feature configurations," in *ICCV*, 2007.

[15] J. Kleban, X. Xie, and W.-Y. Ma, "Spatial pyramid mining for logo detection in natural scenes," in *ICME*, 2008.

[16] A. Gilbert, J. Illingworth, R. Bowden, and G. X. England, "Scale invariant action recognition using compound features mined from dense spatiotemporal corners," in *ECCV*, 2008.

[17] U. M. Khan, B. McCane, and A. Trotman, "Emergent semantic patterns in large scale image dataset: A datamining approach," in *Digital Image Computing Techniques and Applications (DICTA)*. IEEE, 2012.

[18] J. Sivic, B. C. Russell, A. Zisserman, W. T. Freeman, and A. A. Efros, "Unsupervised discovery of visual object class hierarchies," in *CVPR*. IEEE, 2008.

[19] F. Endres, C. Plagemann, C. Stachniss, and W. Burgard, "Unsupervised discovery of object classes from range data using latent dirichlet allocation." in *Robotics: Science and Systems*. The MIT Press, 2009.

[20] K. Grauman and T. Darrell, "Unsupervised learning of categories from sets of partially matching image features," in *CVPR*, 2006.

[21] G. Kim, C. Faloutsos, and M. Hebert, "Unsupervised modeling of object categories using link analysis techniques," in *CVPR*. IEEE, 2008.

[22] M. Bastian, S. Heymann, and M. Jacomy, "Gephi: An open source software for exploring and manipulating networks," 2009.

[23] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997.

[24] S. Martin, W. M. Brown, R. Klavans, and K. W. Boyack, "Openord: an open-source toolbox for large graph layout," in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2011.